

시뮬레이션

소스 코드

최백준 choi@startlink.io



C++14

```
1 #include <iostream>
2 using namespace std;
3 int a[20][20];
4 int dice[7];
5 int dx[] = {0,0,-1,1};
6 int dy[] = {1,-1,0,0};
7 int main() {
8     int n, m, x, y, l;
9     cin >> n >> m >> x >> y >> l;
10    for (int i=0; i<n; i++) {
11        for (int j=0; j<m; j++) {
12            cin >> a[i][j];
13        }
14    }
15    while (l-->0) {
16        int k;
17        cin >> k;
18        k -= 1;
19        int nx, ny;
20        nx = x+dx[k];
21        ny = y+dy[k];
22        if (nx < 0 || nx >= n || ny < 0 || ny >= m) {
23            // 바깥으로 이동시키려고 하는 경우에는 해당 명령을 무시해야 함, 출력도 하면 안됨
24            continue;
25        }
26        if (k == 0) { // right
27            int temp = dice[1];
28            dice[1] = dice[4];
29            dice[4] = dice[6];
30            dice[6] = dice[3];
31            dice[3] = temp;
32        } else if (k == 1) { // left
33            int temp = dice[1];
34            dice[1] = dice[3];
35            dice[3] = dice[6];
36            dice[6] = dice[4];
37            dice[4] = temp;
38        } else if (k == 2) { // up
39            int temp = dice[1];
40            dice[1] = dice[5];
41            dice[5] = dice[6];
42            dice[6] = dice[2];
43            dice[2] = temp;
44        } else { // down
45            int temp = dice[1];
46            dice[1] = dice[2];
47            dice[2] = dice[6];
48            dice[6] = dice[5];
49            dice[5] = temp;
50        }
51        x = nx;
52        y = ny;
53        if (a[x][y] == 0) {
54            // 주사위를 굴렀을 때, 이동한 칸에 써 있는 수가 0이면, 주사위의 바닥면에 써 있는 수가 칸에 복사됨
55            a[x][y] = dice[6];
56        } else {
57            // 0이 아닌 경우에는 칸에 써 있는 수가 주사위의 바닥면으로 복사되며,
58            dice[6] = a[x][y];
59            // 칸에 써 있는 수는 0이 복사됨
60            a[x][y] = 0;
61        }
62        cout << dice[1] << '\n';
63    }
64    return 0;
65 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1988 KB	36 ms	1925 B

Java

```
1 import java.util.*;
2 public class Main {
3     static int[] dx = {0,0,-1,1};
4     static int[] dy = {1,-1,0,0};
5     public static void main(String args[]) {
6         Scanner sc = new Scanner(System.in);
7         int n = sc.nextInt();
8         int m = sc.nextInt();
9         int x = sc.nextInt();
10        int y = sc.nextInt();
11        int l = sc.nextInt();
12        int[][] a = new int[n][m];
13        for (int i=0; i<n; i++) {
14            for (int j=0; j<m; j++) {
15                a[i][j] = sc.nextInt();
16            }
17        }
18        int[] dice = new int[7];
19        while (l-->0) {
20            int k = sc.nextInt() - 1;
21            int nx = x+dx[k];
22            int ny = y+dy[k];
23            if (nx < 0 || nx >= n || ny < 0 || ny >= m) {
24                // 바깥으로 이동시키려고 하는 경우에는 해당 명령을 무시해야 함, 출력도 하면 안됨
25                continue;
26            }
27            if (k == 0) { // right
28                int temp = dice[1];
29                dice[1] = dice[4];
30                dice[4] = dice[6];
31                dice[6] = dice[3];
32                dice[3] = temp;
33            } else if (k == 1) { // left
34                int temp = dice[1];
35                dice[1] = dice[3];
36                dice[3] = dice[6];
37                dice[6] = dice[4];
38                dice[4] = temp;
39            } else if (k == 2) { // up
40                int temp = dice[1];
41                dice[1] = dice[5];
42                dice[5] = dice[6];
43                dice[6] = dice[2];
44                dice[2] = temp;
45            } else { // down
46                int temp = dice[1];
47                dice[1] = dice[2];
48                dice[2] = dice[6];
49                dice[6] = dice[5];
50                dice[5] = temp;
51            }
52            x = nx;
53            y = ny;
54            if (a[x][y] == 0) {
55                // 주사위를 굴렀을 때, 이동한 칸에 써 있는 수가 0이면, 주사위의 바닥면에 써 있는 수가 칸에 복사됨
56                a[x][y] = dice[6];
57            } else {
58                // 0이 아닌 경우에는 칸에 써 있는 수가 주사위의 바닥면으로 복사되며,
59                dice[6] = a[x][y];
60                // 칸에 써 있는 수는 0이 복사됨
61                a[x][y] = 0;
62            }
63            System.out.println(dice[1]);
64        }
65    }
66 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	16724 KB	228 ms	2342 B

C++14

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4 int main() {
5     int n = 4;
6     vector<string> a(n);
7     for (int i=0; i<n; i++) {
8         cin >> a[i];
9     }
10    int k;
11    cin >> k;
12    while (k-->0) {
13        int no, dir;
14        cin >> no >> dir;
15        no -= 1;
16        // 각각의 톱니는 동시에 회전해야 하기 때문에
17        // 먼저, 각 톱니가 어떤 방향으로 회전해야 하는지 구하자
18        vector<int> d(n);
19        d[no] = dir;
20        // 왼쪽 톱니를 연속적으로 구하고
21        for (int i=no-1; i>=0; i--) {
22            if (a[i][2] != a[i+1][6]) {
23                d[i] = -d[i+1];
24            } else {
25                // 한 톱니가 회전하지 않으면
26                // 그 톱니의 왼쪽에 있는 톱니도 회전하지 않는다.
27                break;
28            }
29        }
30        // 오른쪽 톱니를 연속적으로 구하고
31        for (int i=no+1; i<n; i++) {
32            if (a[i-1][2] != a[i][6]) {
33                d[i] = -d[i-1];
34            } else {
35                // 한 톱니가 회전하지 않으면
36                // 그 톱니의 오른쪽에 있는 톱니도 회전하지 않는다.
37                break;
38            }
39        }
40        for (int i=0; i<n; i++) {
41            if (d[i] == 0) continue;
42            if (d[i] == 1) {
43                // 시계 방향 회전
44                char temp = a[i][7];
45                for (int j=7; j>=1; j--) {
46                    a[i][j] = a[i][j-1];
47                }
48                a[i][0] = temp;
49            } else if (d[i] == -1) {
50                // 반시계 방향 회전
51                char temp = a[i][0];
52                for (int j=0; j<7; j++) {
53                    a[i][j] = a[i][j+1];
54                }
55                a[i][7] = temp;
56            }
57        }
58        int ans = 0;
59        for (int i=0; i<n; i++) {
60            if (a[i][0] == '1') {
61                ans |= (1 << i);
62            }
63        }
64    }
65    cout << ans << '\n';
66    return 0;
67 }
```

Handwritten notes and diagrams:

- Top right: "각 톱니" (Each gear) with a circled "톱니" (Gear).
- Below line 12: Four circles representing gears.
- Between lines 18-20: A diagram showing a gear (no) with arrows pointing to its neighbors (i+1 and i-1) to determine their rotation direction.
- Between lines 21-23: A diagram showing a gear (no) with arrows pointing to its neighbors (i+1 and i-1) to determine their rotation direction.
- Between lines 31-33: A diagram showing a gear (no) with arrows pointing to its neighbors (i+1 and i-1) to determine their rotation direction.
- Between lines 41-43: A diagram showing a gear (no) with arrows pointing to its neighbors (i+1 and i-1) to determine their rotation direction.
- Between lines 41-43: A diagram showing a gear (no) with arrows pointing to its neighbors (i+1 and i-1) to determine their rotation direction.
- Between lines 41-43: A diagram showing a gear (no) with arrows pointing to its neighbors (i+1 and i-1) to determine their rotation direction.

결과	메모리	시간	코드 길이
맞았습니다!!!	1988 KB	0 ms	1982 B

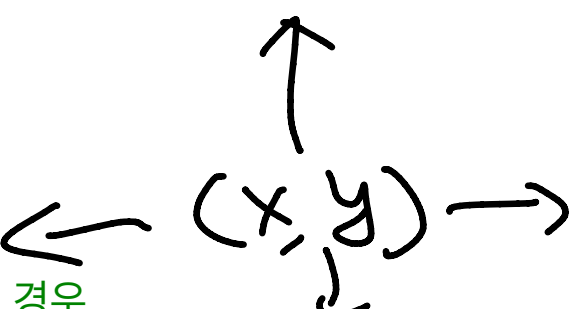
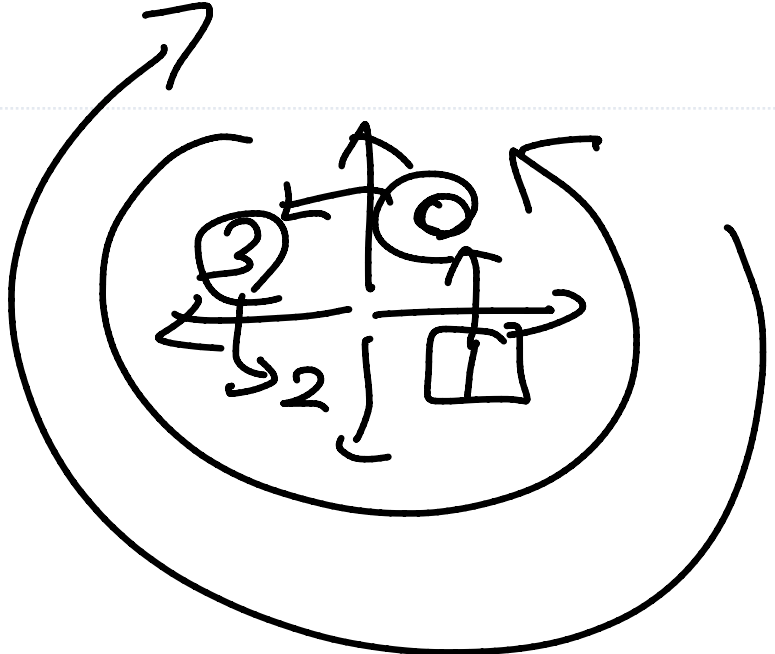
Java

```
1 import java.util.*;
2 public class Main {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int n = 4;
6         char[][] a = new char[4][8];
7         for (int i=0; i<n; i++) {
8             a[i] = sc.next().toCharArray();
9         }
10        int k = sc.nextInt();
11        while (k-->0) {
12            int no = sc.nextInt()-1;
13            int dir = sc.nextInt();
14            // 각각의 톱니는 동시에 회전해야 하기 때문에
15            // 먼저, 각 톱니가 어떤 방향으로 회전해야 하는지 구하자
16            int[] d = new int[n];
17            d[no] = dir;
18            // 왼쪽 톱니를 연속적으로 구하고
19            for (int i=no-1; i>=0; i--) {
20                if (a[i][2] != a[i+1][6]) {
21                    d[i] = -d[i+1];
22                } else {
23                    // 한 톱니가 회전하지 않으면
24                    // 그 톱니의 왼쪽에 있는 톱니도 회전하지 않는다.
25                    break;
26                }
27            }
28            // 오른쪽 톱니를 연속적으로 구하고
29            for (int i=no+1; i<n; i++) {
30                if (a[i-1][2] != a[i][6]) {
31                    d[i] = -d[i-1];
32                } else {
33                    // 한 톱니가 회전하지 않으면
34                    // 그 톱니의 오른쪽에 있는 톱니도 회전하지 않는다.
35                    break;
36                }
37            }
38            for (int i=0; i<n; i++) {
39                if (d[i] == 0) continue;
40                if (d[i] == 1) {
41                    // 시계 방향 회전
42                    char temp = a[i][7];
43                    for (int j=7; j>=1; j--) {
44                        a[i][j] = a[i][j-1];
45                    }
46                    a[i][0] = temp;
47                } else if (d[i] == -1) {
48                    // 반시계 방향 회전
49                    char temp = a[i][0];
50                    for (int j=0; j<7; j++) {
51                        a[i][j] = a[i][j+1];
52                    }
53                    a[i][7] = temp;
54                }
55            }
56        }
57        int ans = 0;
58        for (int i=0; i<n; i++) {
59            if (a[i][0] == '1') {
60                ans |= (1 << i);
61            }
62        }
63        System.out.println(ans);
64    }
65 }
66 }
```

결과	메모리	시간	코드 길이
맞았습니다!!!	11692 KB	120 ms	2310 B

C++14

```
1 #include <iostream>
2 using namespace std;
3 int dx[] = {-1,0,1,0};
4 int dy[] = {0,1,0,-1};
5 int a[55][55];
6 int main() {
7     int n, m;
8     cin >> n >> m;
9     int x, y, dir;
10    cin >> x >> y >> dir;
11    for (int i=0; i<n; i++) {
12        for (int j=0; j<m; j++) {
13            cin >> a[i][j];
14        }
15    }
16    int cnt = 0;
17    // a[x][y] == 0 (청소하지 않은 공간)
18    // a[x][y] == 1 (벽)
19    // a[x][y] == 2 (청소한 공간)
20    while (true) {
21        if (a[x][y] == 0) {
22            // 1. 현재 위치를 청소한다.
23            a[x][y] = 2;
24        }
25        // 2-3, 2-4. 네 방향 모두 청소가 이미 되어있거나 벽인 경우
26        if (a[x+1][y] != 0 && a[x-1][y] != 0 && a[x][y-1] != 0 && a[x][y+1] != 0) {
27            if (a[x-dx[dir]][y-dy[dir]] == 1) {
28                // 2-4. 뒤쪽 방향이 벽이라 후진도 할 수 없는 경우에는 작동을 멈춘다.
29                break;
30            } else {
31                // 2-3. 바라보는 방향을 유지한 채로 한 칸 후진을 하고 2번으로 돌아간다.
32                x -= dx[dir];
33                y -= dy[dir];
34            }
35        } else {
36            // 2-1. 왼쪽 방향에 아직 청소하지 않은 공간이 존재한다면, 그 방향으로 회전한 다음
37            // 2-2. 왼쪽 방향에 청소할 공간이 없다면, 그 방향으로 회전하고 2번으로 돌아간다.
38
39            dir = (dir + 3) % 4;
40            if (a[x+dx[dir]][y+dy[dir]] == 0) {
41                // 2-1. 한 칸을 전진하고 1번부터 진행한다.
42                x += dx[dir];
43                y += dy[dir];
44            }
45        }
46    }
47    for (int i=0; i<n; i++) {
48        for (int j=0; j<m; j++) {
49            if (a[i][j] == 2) {
50                cnt += 1;
51            }
52        }
53    }
54    cout << cnt << '\n';
55    return 0;
56 }
```



$$(dir-1) \% 4$$
$$-1 \% 4 = -1$$

결과	메모리	시간	코드 길이
맞았습니다!!	2000 KB	0 ms	1825 B

Java

```
1 import java.util.*;
2 public class Main {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         int n = sc.nextInt();
6         int m = sc.nextInt();
7         int[][] a = new int[n][m];
8         int x = sc.nextInt();
9         int y = sc.nextInt();
10        int dir = sc.nextInt();
11        for (int i=0; i<n; i++) {
12            for (int j=0; j<m; j++) {
13                a[i][j] = sc.nextInt();
14            }
15        }
16        int cnt = 0;
17        int[] dx = {-1,0,1,0};
18        int[] dy = {0,1,0,-1};
19        // a[x][y] == 0 (청소하지 않은 공간)
20        // a[x][y] == 1 (벽)
21        // a[x][y] == 2 (청소한 공간)
22        while (true) {
23            if (a[x][y] == 0) {
24                // 1. 현재 위치를 청소한다.
25                a[x][y] = 2;
26            }
27            // 2-3, 2-4. 네 방향 모두 청소가 이미 되어있거나 벽인 경우
28            if (a[x+1][y] != 0 && a[x-1][y] != 0 && a[x][y-1] != 0 && a[x][y+1] != 0) {
29                if (a[x-dx[dir]][y-dy[dir]] == 1) {
30                    // 2-4. 뒤쪽 방향이 벽이라 후진도 할 수 없는 경우에는 작동을 멈춘다.
31                    break;
32                } else {
33                    // 2-3. 바라보는 방향을 유지한 채로 한 칸 후진을 하고 2번으로 돌아간다.
34                    x -= dx[dir];
35                    y -= dy[dir];
36                }
37            } else {
38                // 2-1. 왼쪽 방향에 아직 청소하지 않은 공간이 존재한다면, 그 방향으로 회전한 다음
39                // 2-2. 왼쪽 방향에 청소할 공간이 없다면, 그 방향으로 회전하고 2번으로 돌아간다.
40                dir = (dir + 3) % 4;
41                if (a[x+dx[dir]][y+dy[dir]] == 0) {
42                    // 2-1. 한 칸을 전진하고 1번부터 진행한다.
43                    x += dx[dir];
44                    y += dy[dir];
45                }
46            }
47        }
48        for (int i=0; i<n; i++) {
49            for (int j=0; j<m; j++) {
50                if (a[i][j] == 2) {
51                    cnt += 1;
52                }
53            }
54        }
55        System.out.println(cnt);
56    }
57 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	15728 KB	164 ms	2188 B

C++14

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4 bool go(vector<int> &a, int l) {
5     // 한 줄 검사
6     int n = a.size();
7     vector<bool> c(n, false);
8     for (int i=1; i<n; i++) {
9         if (a[i-1] != a[i]) {
10             // 인접한 칸의 높이가 다르면, 경사로를 놓아야 한다.
11             int diff = a[i]-a[i-1];
12             if (diff < 0) diff = -diff;
13             if (diff != 1) {
14                 // 낮은 칸과 높은 칸의 높이 차이는 1이어야 한다.
15                 return false;
16             }
17             if (a[i-1] < a[i]) {
18                 for (int j=1; j<=l; j++) {
19                     if (i-j < 0) {
20                         // 경사로를 놓다가 범위를 벗어나는 경우
21                         return false;
22                     }
23                     if (a[i-1] != a[i-j]) {
24                         // 낮은 지점의 칸의 높이가 모두 같지 않거나, L개가 연속되지 않은 경우
25
26                         return false;
27                     }
28                     if (c[i-j]) {
29                         // 경사로를 놓은 곳에 또 경사로를 놓는 경우
30                         return false;
31                     }
32                     c[i-j] = true;
33                 }
34             } else {
35                 // a[i-1] > a[i]
36                 for (int j=0; j<l; j++) {
37                     if (i+j >= n) {
38                         // 경사로를 놓다가 범위를 벗어나는 경우
39                         return false;
40                     }
41                     if (a[i] != a[i+j]) {
42                         // 낮은 지점의 칸의 높이가 모두 같지 않거나, L개가 연속되지 않은 경우
43
44                         return false;
45                     }
46                     if (c[i+j]) {
47                         // 경사로를 놓은 곳에 또 경사로를 놓는 경우
48                         return false;
49                     }
50                     c[i+j] = true;
51                 }
52             }
53         }
54     }
55     return true;
56 }
57 int main() {
58     int n, l;
59     cin >> n >> l;
60     vector<vector<int>> a(n, vector<int>(n));
61     for (int i=0; i<n; i++) {
62         for (int j=0; j<n; j++) {
63             cin >> a[i][j];
64         }
65     }
66     int ans = 0;
67     for (int i=0; i<n; i++) {
68         // 행 검사
69         vector<int> d;
70         for (int j=0; j<n; j++) {
71             d.push_back(a[i][j]);
72         }
73         if (go(d, l)) ans += 1;
74     }
75     for (int j=0; j<n; j++) {
76         // 열 검사
77         vector<int> d;
78         for (int i=0; i<n; i++) {
79             d.push_back(a[i][j]);
80         }
81         if (go(d, l)) ans += 1;
82     }
83     cout << ans << '\n';
84     return 0;
85 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1988 KB	0 ms	2667 B

Java

```
1 import java.util.*;
2 public class Main {
3     static boolean go(int[] a, int l) {
4         // 한 줄 검사
5         int n = a.length;
6         boolean[] c = new boolean[n];
7         for (int i=1; i<n; i++) {
8             if (a[i-1] != a[i]) {
9                 // 인접한 칸의 높이가 다르면, 경사로를 놓아야 한다.
10                 int diff = a[i]-a[i-1];
11                 if (diff < 0) diff = -diff;
12                 if (diff != 1) {
13                     // 낮은 칸과 높은 칸의 높이 차이는 1이어야 한다.
14                     return false;
15                 }
16                 if (a[i-1] < a[i]) {
17                     for (int j=1; j<=l; j++) {
18                         if (i-j < 0) {
19                             // 경사로를 놓다가 범위를 벗어나는 경우
20                             return false;
21                         }
22                         if (a[i-1] != a[i-j]) {
23                             // 낮은 지점의 칸의 높이가 모두 같지 않거나, L개가 연속되지 않은 경우
24
25                             return false;
26                         }
27                         if (c[i-j]) {
28                             // 경사로를 놓은 곳에 또 경사로를 놓는 경우
29                             return false;
30                         }
31                         c[i-j] = true;
32                     }
33                 } else {
34                     // a[i-1] > a[i]
35                     for (int j=0; j<l; j++) {
36                         if (i+j >= n) {
37                             // 경사로를 놓다가 범위를 벗어나는 경우
38                             return false;
39                         }
40                         if (a[i] != a[i+j]) {
41                             // 낮은 지점의 칸의 높이가 모두 같지 않거나, L개가 연속되지 않은 경우
42
43                             return false;
44                         }
45                         if (c[i+j]) {
46                             // 경사로를 놓은 곳에 또 경사로를 놓는 경우
47                             return false;
48                         }
49                         c[i+j] = true;
50                     }
51                 }
52             }
53         }
54         return true;
55     }
56     public static void main(String[] args) {
57         Scanner sc = new Scanner(System.in);
58         int n = sc.nextInt();
59         int l = sc.nextInt();
60         int[][] a = new int[n][n];
61         for (int i=0; i<n; i++) {
62             for (int j=0; j<n; j++) {
63                 a[i][j] = sc.nextInt();
64             }
65         }
66         int ans = 0;
67         for (int i=0; i<n; i++) {
68             // 행 검사
69             int[] d = new int[n];
70             for (int j=0; j<n; j++) {
71                 d[j] = a[i][j];
72             }
73             if (go(d, l)) ans += 1;
74         }
75         for (int j=0; j<n; j++) {
76             // 열 검사
77             int[] d = new int[n];
78             for (int i=0; i<n; i++) {
79                 d[i] = a[i][j];
80             }
81             if (go(d, l)) ans += 1;
82         }
83         System.out.println(ans);
84     }
85 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	22844 KB	248 ms	3050 B

C++14

```
1 #include <iostream>
2 #include <cstring>
3 using namespace std;
4 int d[100][100];
5 bool apple[100][100];
6 int dx[] = {0,1,0,-1};
7 int dy[] = {1,0,-1,0};
8 int main() {
9     int n;
10    cin >> n;
11    int m;
12    cin >> m;
13    while (m-->0) {
14        int x, y;
15        cin >> x >> y;
16        apple[x-1][y-1] = true;
17    }
18    memset(d,-1,sizeof(d));
19    int x = 0;
20    int y = 0;
21    int dir = 0;
22    int len = 1;
23    d[x][y] = 0;
24    cin >> m;
25    int now = 0;
26    for (int k=0; k<=m; k++) {
27        int t = n*n+1;
28        char ch = 'L';
29        if (k < m) {
30            cin >> t >> ch;
31        }
32        while (now < t) {
33            now += 1;
34            x += dx[dir];
35            y += dy[dir];
36            if (x < 0 || x >= n || y < 0 || y >= n) {
37                cout << now << '\n';
38                return 0;
39            }
40            if (apple[x][y]) {
41                apple[x][y] = false;
42                len += 1;
43            }
44            if (d[x][y] != -1 && now-d[x][y] <= len) {
45                cout << now << '\n';
46                return 0;
47            }
48            d[x][y] = now;
49        }
50        if (ch == 'L') {
51            dir = (dir + 3) % 4;
52        } else {
53            dir = (dir + 1) % 4;
54        }
55    }
56    return 0;
57 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	2036 KB	0 ms	1229 B

Java

```
1 import java.util.*;
2 public class Main {
3     static final int[] dx = {0,1,0,-1};
4     static final int[] dy = {1,0,-1,0};
5     public static void main(String[] args) {
6         Scanner sc = new Scanner(System.in);
7         int n = sc.nextInt();
8         int m = sc.nextInt();
9         int[][] d = new int[n][n];
10        boolean[][] apple = new boolean[n][n];
11        while (m-- > 0) {
12            int x = sc.nextInt()-1;
13            int y = sc.nextInt()-1;
14            apple[x][y] = true;
15        }
16        for (int i=0; i<n; i++) {
17            Arrays.fill(d[i],-1);
18        }
19        int x = 0;
20        int y = 0;
21        int dir = 0;
22        int len = 1;
23        d[x][y] = 0;
24        m = sc.nextInt();
25        int now = 0;
26        for (int k=0; k<=m; k++) {
27            int t = n*n+1;
28            char ch = 'L';
29            if (k < m) {
30                t = sc.nextInt();
31                ch = sc.next().charAt(0);
32            }
33            while (now < t) {
34                now += 1;
35                x += dx[dir];
36                y += dy[dir];
37                if (x < 0 || x >= n || y < 0 || y >= n) {
38                    System.out.println(now);
39                    System.exit(0);
40                }
41                if (apple[x][y]) {
42                    apple[x][y] = false;
43                    len += 1;
44                }
45                if (d[x][y] != -1 && now-d[x][y] <= len) {
46                    System.out.println(now);
47                    System.exit(0);
48                }
49                d[x][y] = now;
50            }
51            if (ch == 'L') {
52                dir = (dir + 3) % 4;
53            } else {
54                dir = (dir + 1) % 4;
55            }
56        }
57    }
58 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	13280 KB	148 ms	1678 B

C++14

```
1 #include <iostream>
2 #include <vector>
3 #include <algorithm>
4 using namespace std;
5 bool c[101][101];
6 int dx[] = {0,-1,0,1};
7 int dy[] = {1,0,-1,0};
8 vector<int> curve(int x, int y, int dir, int gen) {
9     vector<int> ans = {dir};
10    for (int g=1; g<=gen; g++) {
11        vector<int> temp(ans);
12        reverse(temp.begin(), temp.end());
13        for (int &i : temp) {
14            i = (i+1)%4;
15        }
16        ans.insert(ans.end(), temp.begin(), temp.end());
17    }
18    return ans;
19 }
20 int main() {
21     int n;
22     cin >> n;
23     while (n-->0) {
24         int x, y, dir, gen;
25         cin >> y >> x >> dir >> gen;
26         vector<int> dirs = curve(x,y,dir,gen);
27         c[x][y] = true;
28         for (int d : dirs) {
29             x += dx[d];
30             y += dy[d];
31             c[x][y] = true;
32         }
33     }
34     int ans = 0;
35     for (int i=0; i<=99; i++) {
36         for (int j=0; j<=99; j++) {
37             if (c[i][j] && c[i][j+1] && c[i+1][j] && c[i+1][j+1]) {
38                 ans += 1;
39             }
40         }
41     }
42     cout << ans << '\n';
43     return 0;
44 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1996 KB	0 ms	1040 B

Java

```
1 import java.util.*;
2 public class Main {
3     static boolean[][] c = new boolean[101][101];
4     static int[] dx = {0,-1,0,1};
5     static int[] dy = {1,0,-1,0};
6     static ArrayList<Integer> curve(int x, int y, int dir, int gen) {
7         ArrayList<Integer> ans = new ArrayList<>();
8         ans.add(dir);
9         for (int g=1; g<=gen; g++) {
10             ArrayList<Integer> temp = new ArrayList<>(ans);
11             Collections.reverse(temp);
12             for (int i=0; i<temp.size(); i++) {
13                 temp.set(i, (temp.get(i)+1)%4);
14             }
15             ans.addAll(temp);
16         }
17         return ans;
18     }
19     public static void main(String[] args) {
20         Scanner sc = new Scanner(System.in);
21         int n = sc.nextInt();
22         while (n-->0) {
23             int y = sc.nextInt();
24             int x = sc.nextInt();
25             int dir = sc.nextInt();
26             int gen = sc.nextInt();
27             ArrayList<Integer> dirs = curve(x,y,dir,gen);
28             c[x][y] = true;
29             for (int d : dirs) {
30                 x += dx[d];
31                 y += dy[d];
32                 c[x][y] = true;
33             }
34         }
35         int ans = 0;
36         for (int i=0; i<=99; i++) {
37             for (int j=0; j<=99; j++) {
38                 if (c[i][j] && c[i][j+1] && c[i+1][j] && c[i+1][j+1]) {
39                     ans += 1;
40                 }
41             }
42         }
43         System.out.println(ans);
44     }
45 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	11736 KB	136 ms	1412 B

C++14

```
1 #include <iostream>
2 #include <cstring>
3 #include <algorithm>
4 #include <queue>
5 #include <vector>
6 using namespace std;
7 char a[111][111];
8 int c[111][111];
9 int n, m;
10 vector<pair<int,int>> group;
11 int dx[] = {0,0,1,-1};
12 int dy[] = {1,-1,0,0};
13 void dfs(int x, int y) {
14     if (a[x][y] == '.') return;
15     if (c[x][y]) return;
16     c[x][y] = true;
17     group.push_back(make_pair(x,y));
18     for (int k=0; k<4; k++) {
19         int nx = x+dx[k];
20         int ny = y+dy[k];
21         if (0 <= nx && nx < n && 0 <= ny && ny < m) {
22             dfs(nx,ny);
23         }
24     }
25 }
26 void simulate() {
27     memset(c, false, sizeof(c));
28     for (int i=0; i<n; i++) {
29         for (int j=0; j<m; j++) {
30             if (a[i][j] == '.') continue;
31             if (c[i][j]) continue;
32             group.clear();
33             dfs(i,j);
34             vector<int> low(m, -1);
35             for (auto &p : group) {
36                 low[p.second] = max(low[p.second], p.first);
37                 a[p.first][p.second] = '.';
38             }
39             int lowest = n;
40             for (int i, j=0; j<m; j++) {
41                 if (low[j] == -1) continue;
42                 for (i=low[j]; i<n && a[i][j]!='.'; i++);
43                 lowest = min(lowest, i-low[j]-1);
44             }
45             for (auto &p : group) {
46                 p.first += lowest;
47                 a[p.first][p.second] = 'x';
48                 c[p.first][p.second] = true;
49             }
50         }
51     }
52 }
53 int main() {
54     cin >> n >> m;
55     for (int i=0; i<n; i++) {
56         cin >> a[i];
57     }
58     int k;
59     cin >> k;
60     for (int i=0; i<k; i++) {
61         int height;
62         cin >> height;
63         height = n - height;
64         if (i%2 == 0) {
65             for (int j=0; j<m; j++) {
66                 if (a[height][j] == 'x') {
67                     a[height][j] = '.';
68                     break;
69                 }
70             }
71         } else {
72             for (int j=m-1; j>=0; j--) {
73                 if (a[height][j] == 'x') {
74                     a[height][j] = '.';
75                     break;
76                 }
77             }
78         }
79         simulate();
80     }
81     for (int i=0; i<n; i++) {
82         cout << a[i] << '\n';
83     }
84     return 0;
85 }
86
```

결과	메모리	시간	코드 길이
맞았습니다!!!	2180 KB	24 ms	2235 B

Java

```
1 import java.util.*;
2 class Pair {
3     int first, second;
4     Pair(int first, int second) {
5         this.first = first;
6         this.second = second;
7     }
8 }
9 public class Main {
10     static char[][] a;
11     static boolean[][] c;
12     static int n, m;
13     static ArrayList<Pair> group = new ArrayList<>();
14     static int[] dx = {0,0,1,-1};
15     static int[] dy = {1,-1,0,0};
16     static void dfs(int x, int y) {
17         if (a[x][y] == '.') return;
18         if (c[x][y]) return;
19         c[x][y] = true;
20         group.add(new Pair(x, y));
21         for (int k=0; k<4; k++) {
22             int nx = x+dx[k];
23             int ny = y+dy[k];
24             if (0 <= nx && nx < n && 0 <= ny && ny < m) {
25                 dfs(nx,ny);
26             }
27         }
28     }
29     static void simulate() {
30         for (int i=0; i<n; i++) {
31             Arrays.fill(c[i], false);
32         }
33         for (int i=0; i<n; i++) {
34             for (int j=0; j<m; j++) {
35                 if (a[i][j] == '.') continue;
36                 if (c[i][j]) continue;
37                 group.clear();
38                 dfs(i,j);
39                 int[] low = new int[m];
40                 Arrays.fill(low, -1);
41                 for (Pair p : group) {
42                     low[p.second] = Math.max(low[p.second], p.first);
43                     a[p.first][p.second] = '.';
44                 }
45                 int lowest = n;
46                 for (int x, y=0; y<m; y++) {
47                     if (low[y] == -1) continue;
48                     for (x=low[y]; x<n && a[x][y]!='.'; x++);
49                     lowest = Math.min(lowest, x-low[y]-1);
50                 }
51                 for (Pair p : group) {
52                     p.first += lowest;
53                     a[p.first][p.second] = 'x';
54                     c[p.first][p.second] = true;
55                 }
56             }
57         }
58     }
59     public static void main(String args[]) {
60         Scanner sc = new Scanner(System.in);
61         n = sc.nextInt();
62         m = sc.nextInt();
63         a = new char[n][m];
64         c = new boolean[n][m];
65         for (int i=0; i<n; i++) {
66             a[i] = sc.next().toCharArray();
67         }
68         int k = sc.nextInt();
69         for (int i=0; i<k; i++) {
70             int height = n - sc.nextInt();
71             if (i%2 == 0) {
72                 for (int j=0; j<m; j++) {
73                     if (a[height][j] == 'x') {
74                         a[height][j] = '.';
75                         break;
76                     }
77                 }
78             } else {
79                 for (int j=m-1; j>=0; j--) {
80                     if (a[height][j] == 'x') {
81                         a[height][j] = '.';
82                         break;
83                     }
84                 }
85             }
86             simulate();
87         }
88         for (int i=0; i<n; i++) {
89             System.out.println(a[i]);
90         }
91     }
92 }
93 }
94
```

결과	메모리	시간	코드 길이
맞았습니다!!!	32624 KB	500 ms	2890 B

C++14

```
1 #include <iostream>
2 #include <string>
3 #include <vector>
4 using namespace std;
5 int main() {
6     string s;
7     getline(cin, s);
8     vector<string> a;
9     string now = "";
10    for (char x : s) {
11        if (x == ' ' || x == ',' || x == ';') {
12            if (now.length() > 0) {
13                a.push_back(now);
14            }
15            now = "";
16        } else {
17            now += x;
18        }
19    }
20    string base = a[0];
21    for (int i=1; i<a.size(); i++) {
22        string t = base;
23        string s = a[i];
24        while (s.length() > 0 && !('a' <= s.back() && s.back() <= 'z')) {
25            if (s.back() == '[') {
26                t += ']';
27            } else if (s.back() == ']') {
28                t += '[';
29            } else {
30                t += s.back();
31            }
32            s.pop_back();
33        }
34        cout << t << ' ' << s << ';' << '\n';
35    }
36    return 0;
37 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	1992 KB	0 ms	922 B

Java

```
1 import java.util.*;
2 public class Main {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         String temp = sc.nextLine();
6         ArrayList<String> a = new ArrayList<>();
7         String now = "";
8         for (char x : temp.toCharArray()) {
9             if (x == ' ' || x == ',' || x == ';') {
10                 if (now.length() > 0) {
11                     a.add(now);
12                 }
13                 now = "";
14             } else {
15                 now += x;
16             }
17         }
18         String base = a.get(0);
19         for (int i=1; i<a.size(); i++) {
20             String t = base;
21             String s = a.get(i);
22             while (s.length() > 0 && !('a' <= s.charAt(s.length()-1) && s.charAt(s.length()-1) <= 'z')) {
23                 char ch = s.charAt(s.length()-1);
24                 if (ch == '[') {
25                     t += ']';
26                 } else if (ch == ']') {
27                     t += '[';
28                 } else {
29                     t += ch;
30                 }
31                 s = s.substring(0, s.length()-1);
32             }
33             System.out.println(t + " " + s + ";");
34         }
35     }
36 }
37 }
```

결과	메모리	시간	코드 길이
맞았습니다!!	11480 KB	104 ms	1189 B

C++14

```
1 #include <iostream>
2 #include <string>
3 using namespace std;
4 int c[10][10]={
5     {1,1,1,0,1,1,1},
6     {0,0,1,0,0,1,0},
7     {1,0,1,1,1,0,1},
8     {1,0,1,1,0,1,1},
9     {0,1,1,1,0,1,0},
10    {1,1,0,1,0,1,1},
11    {1,1,0,1,1,1,1},
12    {1,0,1,0,0,1,0},
13    {1,1,1,1,1,1,1},
14    {1,1,1,1,0,1,1}
15 };
16 int main() {
17     int s;
18     string n;
19     cin >> s >> n;
20     int m = n.length();
21     for (int i=0; i<5; i++) {
22         if (i == 0 || i == 2 || i == 4) {
23             for (int j=0; j<m; j++) {
24                 int now = n[j]-'0';
25                 if (j != 0) {
26                     cout << ' ';
27                 }
28                 cout << ' ';
29                 if ((i == 0 && c[now][0]) || (i == 2 && c[now][3]) || (i == 4 && c[now][6])) {
30                     for (int k=0; k<s; k++) {
31                         cout << (-);
32                     }
33                 } else {
34                     for (int k=0; k<s; k++) {
35                         cout << ' ';
36                     }
37                 }
38                 cout << ' ';
39             }
40             cout << '\n';
41         } else {
42             for (int l=0; l<s; l++) {
43                 for (int j=0; j<m; j++) {
44                     int now = n[j] - '0';
45                     if (j != 0) {
46                         cout << ' ';
47                     }
48                     if ((i == 1 && c[now][1]) || (i == 3 && c[now][4])) {
49                         cout << '|';
50                     } else {
51                         cout << ' ';
52                     }
53                     for (int k=0; k<s; k++) {
54                         cout << ' ';
55                     }
56                     if ((i == 1 && c[now][2]) || (i == 3 && c[now][5])) {
57                         cout << '|';
58                     } else {
59                         cout << ' ';
60                     }
61                 }
62                 cout << '\n';
63             }
64         }
65     }
66     return 0;
67 }
68
```

결과	메모리	시간	코드 길이
맞았습니다!!!	1988 KB	0 ms	1937 B

Java

```
1 import java.util.*;
2 public class Main {
3     static final int[][] c = {
4         {1,1,1,0,1,1,1},
5         {0,0,1,0,0,1,0},
6         {1,0,1,1,1,0,1},
7         {1,0,1,1,0,1,1},
8         {0,1,1,1,0,1,0},
9         {1,1,0,1,0,1,1},
10        {1,1,0,1,1,1,1},
11        {1,0,1,0,0,1,0},
12        {1,1,1,1,1,1,1},
13        {1,1,1,1,0,1,1}
14    };
15     public static void main(String[] args) {
16         Scanner sc = new Scanner(System.in);
17         int s = sc.nextInt();
18         String n = sc.next();
19         int m = n.length();
20         for (int i=0; i<5; i++) {
21             if (i == 0 || i == 2 || i == 4) {
22                 for (int j=0; j<m; j++) {
23                     int now = n.charAt(j)-'0';
24                     if (j != 0) {
25                         System.out.print(" ");
26                     }
27                     System.out.print(" ");
28                     if ((i == 0 && c[now][0] == 1) || (i == 2 && c[now][3] == 1) || (i == 4 && c[now][6] == 1))
29 {
30                     for (int k=0; k<s; k++) {
31                         System.out.print("-");
32                     }
33                 } else {
34                     for (int k=0; k<s; k++) {
35                         System.out.print(" ");
36                     }
37                     System.out.print(" ");
38                 }
39                 System.out.println();
40             } else {
41                 for (int l=0; l<s; l++) {
42                     for (int j=0; j<m; j++) {
43                         int now = n.charAt(j) - '0';
44                         if (j != 0) {
45                             System.out.print(" ");
46                         }
47                         if ((i == 1 && c[now][1] == 1) || (i == 3 && c[now][4] == 1)) {
48                             System.out.print("|");
49                         } else {
50                             System.out.print(" ");
51                         }
52                         for (int k=0; k<s; k++) {
53                             System.out.print(" ");
54                         }
55                         if ((i == 1 && c[now][2] == 1) || (i == 3 && c[now][5] == 1)) {
56                             System.out.print("|");
57                         } else {
58                             System.out.print(" ");
59                         }
60                     }
61                     System.out.println();
62                 }
63             }
64         }
65     }
66 }
67 }
68
```

결과	메모리	시간	코드 길이
맞았습니다!!!	12172 KB	152 ms	2422 B

끝

코드 플러스

<https://code.plus>

- 슬라이드에 포함된 소스 코드를 보려면 "정보 수정 > 백준 온라인 저지 연동"을 통해 연동한 다음, "백준 온라인 저지"에 로그인해야 합니다.
- 강의 내용에 대한 질문은 코드 플러스의 "질문 게시판"에서 할 수 있습니다.
- 문제와 소스 코드는 슬라이드에 첨부된 링크를 통해서 볼 수 있으며, "백준 온라인 저지"에서 서비스됩니다.
- 슬라이드와 동영상 강의는 코드 플러스 사이트를 통해서만 볼 수 있으며, 동영상 강의의 녹화와 다운로드, 배포와 유통은 저작권법에 의해서 금지되어 있습니다.
- 다른 경로로 이 슬라이드나 동영상 강의를 본 경우에는 codeplus@startlink.io 로 이메일 보내주세요.
- 강의 내용, 동영상 강의, 슬라이드, 첨부되어 있는 소스 코드의 저작권은 스타트링크와 최백준에게 있습니다.